A New Review Approach for improving accuracy of Multi Label Stream Data

Kunal Shah
Department of Information Technology
L.D. Engineering College, Gujarat Technological University

Abstract— Many real world problems involve data which can be considered as multi-label data streams. Efficient methods exist for multi-label classification in non streaming scenarios. However, learning in evolving streaming scenarios is more challenging, as the learners must be able to adapt to change using limited time and memory. Classification is used to predict class of unseen instance as accurate as possible. Multi label classification is a variant of single label classification where set of labels associated with single instance. Multi label classification is used by modern applications, such as text classification, functional genomics, image classification, music categorization etc. This paper introduces the task of multi-label classification, methods for multi-label classification and evolution measure for multi-label classification. Also done comparative analysis of multi label classification methods on the basis of theoretical study and then on the basis of simulation done on various data sets.

Key words: Problem Transformation Method, Binary Relevance Method, Multi-label classification

I. INTRODUCTION

Real-time analysis of data streams is becoming a key area of data mining research as the number of applications demanding such processing increases. Nowadays, data is generated at an increasing rate from sensor applications, measurements in network monitoring and traffic management, log records or click-streams in web exploring, manufacturing processes, call detail records, email, blogging, twitter posts, and other sources.

In the traditional supervised classification task, each example is associated with a single class label. A classifier learns to associate each new unseen example with exactly one of these known class labels. When each example may be associated with multiple labels, then this is called multi- label classification. Hence multi-label classification is simply the classification task where each example may be associated with multiple labels.

A classification task usually involves separating data into training and testing sets. Each instance in the training set contains one class label and several attributes. The goal of classifier is to produce a model which predicts label of the test data given only the test data attributes. In classification problems, each instance of a dataset is associated with just one class label that is single label classification. (As shown in fig. 1)

Fig 1: Single Label Classification

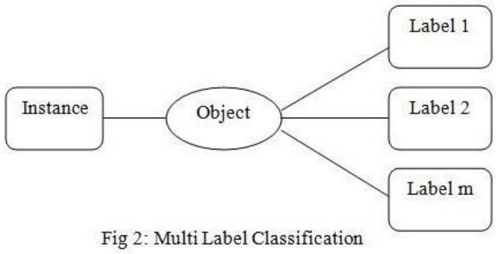

Fig 2: Multi Label Classification

However, there are many classification tasks where each instance can be associated with one or more class labels. This group of problems represents an area known as Multi-Label Classification. (As shown in fig. 2) Multi-label classification methods are increasingly required by modern applications, such as text classification, gene functionality, music categorization and semantic scene classification. The number of class labels is predicted for each instance.

II. MULTILABEL CLASSIFICATION

A complete framework for mining multi-label instances from evolving data streams must have the following components:

- Generators of evolving multi-label streams
- Multi-label adaptive classifiers
- An incremental streaming evaluation component

There are mainly two methods for multi-label classification problems: (1) problem transformation method and (2) algorithm adaptation method. Problem transformation method transfers multi-label problems into single label problems. And algorithm adaptation method extends specific learning algorithm to handle multi-label problems.

Below table shows example of multi label problem, with five class labels. L = {rec, sport, swim, auto, run}

Attributes		Class Labels				
A	**B**	**rec**	**sport**	**swim**	**auto**	**run**
A	1	√	√	√		
A	2	√	√			√
A	2	√	√	√		√
B	1	√	√			
B	2	√			√	

Table 1: Example of Multi Label Problem

III. PROBLEM TRANSFORMATION METHOD

In this method, the main idea is to transfer multi label problem into a set of single label problems. It is an algorithm independent method so any traditional classification algorithm can be used to deal with multi label problems. There are several problem transformation methods available for transferring multi label problems into single label problems. For example, L = {L1, L2, L3, L4} where L is number of labels.

Example	Label Set
1	{L1,L4}
2	{L3,L4}
3	{L1}
4	{L2,L3,L4}

Table 2: Multi Label Example

A. Binary Relevance

This method is basically binary classification of labels. So it transforms original multi label dataset into |L| single label dataset. It builds binary classifier for each label. For the classification of new instance, BR gives union of the labels that are positively predicted by |L| classifier.

A relevant advantage of the BR approach is its low computational complexity compared with other multi label methods. For a constant number of examples, BR scales linearly with size q of the label set L. Considering that the complexity of the base-classifiers is bound to O(C), the complexity of BR is qxO(C). Thus, the BR approach is quite appropriate for not very large q.

As shown in Table 3, BR method gives four individual classifier (|L|=4) from Table 2

EX #	L1
1	√
2	
3	√
4	

EX #	L2
1	
2	
3	
4	√

EX#	L3
1	
2	√
3	
4	√

EX#	L4
1	√
2	√
3	
4	√

Table 3: Binary Relevance Method

B. Ranking via Single Label

This method transforms the multi label dataset into single label dataset. There are different ways for transformation like ignore multi label instance, find maximum count of labels, find minimum count of labels, random selection of label and assign weight to each labels. A single label classifier outputs a vote (probability) for each class label which produce ranking. (As shown if Table 4)

EX#	Label Set
3	{L1}

(a)Ignore

EX#	Label
1	L4
2	L4
3	L1
4	L4

(b)Maximum

EX#	Label
1	L4
2	L4
3	L1
4	L4

(c) Minimum

EX#	Label
1	L4
2	L4
3	L4
4	L1

(d)Random

EX#	Label	Weight
1	L1	0.50
1	L4	0.50
2	L3	0.50
2	L4	0.50
3	L1	1.00
4	L2	0.33
4	L3	0.33
4	L4	0.33

(e)Copy Weight

Table 4: Ranking via single label

C. Ranking Via Pairwise Comparision

This method performs pair wise comparison of labels. It learns m=k(k-1)/2 binary models, one model for each pair of labels. (Where k is number of labels=|L|) Model is trained based on examples that are annotated by at least one of the labels, but not both. So for new instance, all m models are invoked and ranking is obtained by counting the votes received by each label. (see Table 5 and Table 6)

EX #	L1_L2
1	L1
3	L1
4	L2

EX #	L1_L3
1	L1
2	L3
3	L1
4	L3

EX #	L1_L4
2	L4
3	L1
4	L4

EX #	L1_L3
2	L1

EX #	L1_L4
1	L4
2	L4

EX #	L1_v
1	L1
2	V
3	L1
4	V

Table 5: One classifier for each pair of label

New instance x':

L1_L2	L1_L3	L1_L4	L2_L3	L2_L4	L3_L4
L1	L3	L1	L3	L2	L3

Votes for each label:

L1	L2	L3	L4
2	1	3	0

Ranking based on votes:

r(L3) > r(L1) > r(L2) > r(L4)

Table 6: Ranking of labels for new instance

D. Calibrated Label Ranking

This method is extension of RPC method. It introduces one additional virtual label V, with the purpose of separating positive and negative labels. Final ranking is obtained by votes of all labels including virtual label V. (As shown in Table 7).

EX #	L1_V
1	L1
2	V
3	L1
4	V

EX #	L2_V
1	V
2	V
3	V
4	L2

EX #	L3_V
1	V
2	L3
3	V
4	L3

EX #	L4_V
1	L4
2	L4
3	V
4	L4

Table 7: Calibrated ranking of labels

New instance x':

L1_L2	L1_L3	L1_L4	L2_L3	L2_L4	L3_L4
L1	L3	L1	L3	L2	L3

L1_V	L2_V	L3_V	L4_V
L1	V	V	V

Votes for each label:

L1	L2	L3	L4	V
4	2	0	1	3

Ranking based on votes:

r (L1) > r(LV) > r(L2) > r(L4) < r(L3)

Table 8: Ranking of labels for new instance

E. Label Power set

This method replaces each unique subset (Distinct Label Set) of labels that exists in multi label dataset with single label. So LP introduces new set of class labels. For new instance, base classifier of LP predicts one label which is originally a set of labels in multi label dataset. Below Table 9 shows LP method performs on Table 2. For first instance label L1, L2 are present and label L3, L4 are absent so LP gives 1001.

Ex #	Label(L1L2L3L4)
1	1001
2	0011
3	1000
4	0111

Table 9: Label Power Set

F. Pruned Set

This method transforms multi label dataset into single label dataset using LP method. Pruning parameter p (user defined threshold) identifies pruned examples in given multi label dataset. Pruned examples are those whose label set occur less time than pruning parameter p. The PS method identifies less important examples from multi label dataset. As shown in below Table 10 last row is discarded considering pruning parameter 3.

Label-set	Count
L1	16
L2	14
L2,L3	12
L1,L4	8
L3,L4	7
L1,L2,L3	2

Table 10: Pruned set method for p=3

G. Random K-Label Set

This method randomly breaks a large set of labels into a number n of subsets of small size k, called k-label sets. For training of multi label classifier LP method is used, an average decision is calculated for each label in L. And final decision is positive for a given label if the average decision is larger than threshold t. It considers label correlation ship and avoids LP problems.

Method	Merits	Demerits
BR	Simple binary classification and relatively fast.	Does not consider label correlation ship
Ranking via single label	Conceptually Simple	Not dealing well with overlapping of labels
RPC	Flexible method	Consume more prediction time and more memory space
CLR	It deals with pair wise comparison of each label with virtual label and it also provide ranking	It is conceptually expensive method. Unlabeled data is not considered during classification
LP	It considers label correlation ship	Conceptually complex method and leads to over fitting of training data
PS	Run faster and considers label correlation ship	Dependence on predictions of base classifier

RAkEL	Simpler, considers label correlation ship and more predictive capability	Consumes more time and Unlabeled data is not considered during classification

Table 11: Comparative study of problem transformation method

IV. ALGORITHM ADAPTATION METHOD

In this method, single label classifier is extended to develop multi label classifier to handle multi label problems. So this method is an algorithm dependent method. Various algorithm adaptation methods are developed based on different algorithms.

A. Multi-Label Decision Tree (C4.5)

This algorithm is the extension of basic decision tree algorithm for handling multi label data. In basic decision tree algorithm, entropy formula is modified to handle multiple labels.

B. Multi-Layer Neural Network

The multi-label neural network uses the multi layer feed forward neural network as its base algorithm. Adapting neural network algorithm to classify multi-label instances requires three key steps: (1) Creating a new error function that captures the characteristics of multi-label learning. (2) Modify the network to minimize this new error function. (3) Using threshold function to determine an output is in the relevant set of labels.

C. Back Propagation Multi-Label Learning

BPMLL extends basic back-propagation algorithm by introducing a new global error function that captures the characteristics of multi label learning.

D. Multi-Label K Nearest Neighbours

The algorithm is the extension of kNN algorithm. It uses the kNN algorithm independently for each label. It finds the k nearest examples to the test instance and considers those that are labeled with positive and negative. (MLkNN has also the capability of producing a ranking of the labels as an output.)

E. Multi-Label Boosting

These two algorithms are extensions of basic AdaBoost algorithm for handling multi-label data. Hamming loss is reduced using AdaBoost.MH and accuracy is increased using AdaBoost.MR.

Method	Merits	Demerits
C4.5	Easy to learn and more informative attributes are used for splitting decision tree	Does not consider label correlation ship
BPMLL	Provides better generalization capability to learning system	Because of neural network complexity becomes high in training phase
MLkNN	Work well on image and text data.	Unlabeled data is not considered for classification
AdaBoost.MH AdaBoost.MR	Improved accuracy and minimized hamming loss	Unlabeled data is not considered for classification

Table 12: Comparative study of Algorithm Adaptation

V. MULTI-LABEL HOEFFDING TREES

One of the tasks is to design a decision tree learner for extremely large (potentially infinite) datasets. Such a decision tree learner should require each example to be read at most once, and only a small constant time to process it. This will make it possible to directly mine online data sources (i.e., without ever storing the examples), and to build potentially very complex trees with acceptable computational cost. The problem of deciding exactly how many examples are necessary at each node is solved by using a statistical result known as the Hoeffding bound (or additive Chernoff bound). Consider a real-valued random variable r whose range is R (e.g., for a probability the range is one, and for an information gain the range is log c,

where c is the number of classes). Suppose we have made n independent observations of this variable, and computed their mean r.

Hoeffding bound states that, with probability 1- δ, the true mean of the variable is at least r' – ε, where

– Hoeffding Bound(Additive Chernoff Bound)

- r: random variable representing the attribute selection method
- R: range of r
- N: # independent observations
- Mean of r is at least r_{avg} -€, with probability 1 - ∂

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}$$

Hoeffding-based tree learners have been recognized as the most efficient in terms of processing speed per example, although their learning might be slow, which results in lower any-time accuracy at the beginning. The Hoeffding trees tend to be less accurate in situations where several attributes appear to be equally discriminative.

This problem is solved using option trees, which can include option nodes in addition to ordinary split nodes. The main motivation is that introducing option nodes removes the need for selecting the best splitting attribute. Main idea is to introduce options only when splitting decisions are ambiguous, which will avoid excessive and unnecessary tree growth and reduce memory consumption.

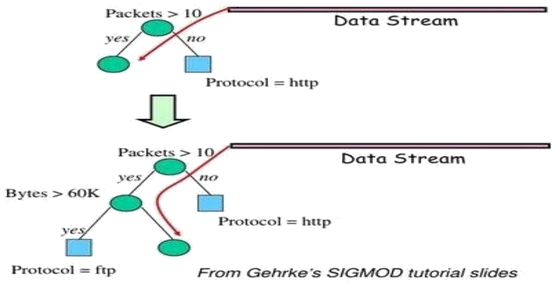

Fig 3: Hoeffding Bound Tree Structure

A. *Option Tress*

Option trees are a single general structure making it possible to travel down multiple paths and arrive at multiple leaves. This is achieved by introducing the possibility of *option nodes* to the tree, alongside the standard decision nodes and leaf nodes. An option node splits the decision path several ways—when an option node is encountered several different sub trees are traversed, which could themselves contain more option nodes, thus the potential for reaching different leaves is multiplied by every option. Making a decision with an option tree involves combining the predictions of the applicable leaves into a final result. A potential benefit of option trees over a traditional ensemble is that the more flexible representation can save space. Consider as an extreme example an ensemble of one hundred mostly identical large trees, where the only difference between each tree lies at a single leaf node, in the same position in each tree. The standard ensemble representation would require one hundred copies of the tree where only the leaf would differ. Efficiently represented as an option tree this would require almost a hundred times less space, where the varying leaf could be replaced by an option node splitting one hundred ways leading to the one hundred different leaf variants.

Adaptive Hoeffding option tree which is quite similar to Hoeffding option tree with several improvements was introduced later for accurate result calculation. Each leaf stores an estimation of the current classification error and the weight of each node in the major voting process which is proportional to the square of the inverse of the error.

VI. RESEARCH CHALLENGES

Following are the research challenges in the field of multi-label classification problem.

– To apply data pre-processing techniques like pruning, feature selection, handle missing value to improve the performance of MLC problem.
– To handle continuous attribute in MLC problem.
– Design a hierarchical structure for multiple labels to manage label correlation ship.
– To extract relevant label set from multiple label set.
– A novel approach is build to use both problem transformation method and algorithm adaptation method for improving performance of multi label classification problem.

VII. CONCLUSION

This paper presented study of different problem transformation methods and algorithm adaptation methods for multi label classification. From comparative study and experimental analysis on different databases concluded that algorithm adaptation method is best option for multi label classification compared to problem transformation method.

Majority class prediction and naive Bayes prediction at the leaves is used for Hoeffding trees and they are evaluated using the framework and the results are noted and accuracy of the algorithms mainly Hoeffding trees and Hoeffding option trees is compared for different data sets under various memory limits. It is observed that option nodes enable faster growth without instability in splitting decisions and have improved lookahead strategy. Accuracy and speed of the algorithms based on Hoeffding option tree and adaptive Hoeffding option tree are observed and graphs are represented for different data sets on various memory limits.

REFERENCES

[1] Purvi Prajapati, Amit Thakkar, Amit Ganatra. “A Survey and Current Research Challenges in Multi-Label Classification Methods”. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012. Pp. 248-252.
[2] Mohammad M. Masud,Clay Woolam ,Jing Gao, Latifur Khan, Jiawei Han, Kevin W, Hamlen, Nikunj C. Oza. “Facing the reality of data stream classification: coping with scarcity of labeled data”. Received: 6 May 2009 / Revised: 26 April 2011 / Accepted: 22 October 2011© Springer-Verlag London Limited 2011.
[3] Jesse Read Albert Bifet Geoff Holmes Bernhard Pfahringer. “Streaming Multi-label Classi_cation”. JMLR: Workshop and Conference Proceedings 17 (2011) 19{25 2nd Workshop on Applications of Pattern Analysis.
[4] Jesse Read Albert Bifet Geoff Holmes Bernhard Pfahringer.” Efficient Multi-label Classification for Evolving Data Streams”. Working Paper Series ISSN 1177-777X.
[5] B. Reshma Yusuf Dr. P. Chenna Reddy.” Mining Data Streams using Option Trees”. Computer Engineering and Intelligent Systems ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online) Vol 3, No.7, 2012.
[6] Xiangnan Kong, Philip S. Yu. “An Ensemble-based Approach to Fast Classification of Multi-label Data Streams”.
[7] Jesse Read, Bernhard Pfahringer, Geoff Holmes. “Generating Synthetic Multi-label Data Streams”.
[8] Grigorios Tsoumakas ,Eleftherios Spyromitros-Xioufis, Jozef Vilcek ,Ioannis Vlahavas.” MULAN: A Java Library for Multi-Label Learning”. Journal of Machine Learning Research 12 (2011) 2411-2414 Submitted 8/09; Revised 7/10; Published 7/11.
[9] Svitlana Volkova.” Data Stream Mining: A Review of Learning Methods and Frameworks”.

[10] Elena Ikonomovska, Suzana Loskovska, Dejan Gjorgjevik.” A SURVEY OF STREAM DATA MINING”. Eighth National Conference with International Participation - ETAI 2007.
[11] Outline:Multi Label Classification http://www.tsc.uc3m.es/~jesse/talks/mend.pdf
[12] Grigorios Tsoumakas, Ioannis Katakis. Multi-Label Classification: An Overview, International Journal of Data Warehousing and Mining, David Taniar (Ed.), Idea Group Publishing, 3(3), pp. 1-13, 2007.
[13] Read, J.; Pfahringer, B.; Holmes, G.; Dept. of Comput. Sci., Univ. of Waikato, Hamilton. A Pruned Problem Transformation Method for Multi-Label Classification. Data Mining, 2008. ICDM '08. Eighth IEEE International Conference, pages: 995 – 1000, 15-19 Dec. 2008
[14] Learning from Multi Label Data http://mlkd.csd.auth.gr/multilabel.html.
[15] [15].Klaus Brinker and Johannes Furnkranz and Eyke Hullermeier. A Unified Model for Multilabel Classification and Ranking. Proceedings of the 2006 conference on ECAI.
[16] [16].Eleftherios Spyromitros Xioufis, Myra Spiliopoulou, Grigorios Tsoumakas, Ioannis Vlahavas. “Dealing with Concept Drift and Class Imbalance in Multi-Label Stream Classification”. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence February 2011.
[17] Everton Alvares Cherman, Maria Carolina Monard, Jean Metz.” Multi-label Problem Transformation Methods: a Case Study”. CLEI ELECTRONIC JOURNAL, VOLUME 14, NUMBER 1, PAPER 4, APRIL 2011.
[18] Jesse Read, Bernhard Pfahringer, Geoff Holmes, Eibe Frank.” Classifier Chains for Multi-label Classification”.
[19] Erica Akemi Tanaka1 and Jose Augusto Baranauskas.” An Adaptation of Binary Relevance for Multi-Label Classification applied to Functional Genomics. Proceedings of the XXXII Congress of the Brazilian Computer Society, XII Workshop on Medical Informatics, Curitiba, PR, Brazil, July 16-19, 2012, ISSN 2175-2761”.
[20] S. Godbole and S. Sarawagi.”Discriminative methods for multi-labeled classification”. In Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 22–30, 2004.